1992

Integrated Simulation for Rapid Development of Autonomous Underwater Vehicles

Brutzman, Donald P.

INTEGRATED SIMULATION FOR RAPID DEVELOPMENT OF AUTONOMOUS UNDERWATER VEHICLES

Donald P. Brutzman, Yutaka Kanayama and Michael J. Zyda
Computer Science Department, Naval Postgraduate School
Monterey California 93943 USA
brutzman@taurus.cs.nps.navy.mil
(408) 646-2174 work, (408) 646-2814 fax

Abstract

The development and testing of Autonomous Underwater Vehicle (AUV) hardware and software is greatly complicated by vehicle inaccessibility during operation. Integrated simulation remotely links vehicle components and support equipment with graphics simulation workstations, allowing complete real-time, pre-mission, pseudo-mission and post-mission visualization and analysis in the lab environment. Integrated simulator testing of AUV software and hardware is a broad and versatile method that supports rapid diagnosis and robust correction of system faults.

Pre-mission simulator AUV testing permits experimental evaluation of developmental software. Pseudo-mission simulator testing of AUV processes employs an identical laboratory microprocessor or remote communication with a testbench-mounted operating AUV, permitting end-to-end testing of all software and hardware. Post-mission simulator playback of recorded telemetry, sensor data and system state transitions supports in-depth reenactment, playback and analysis of in-water operational results.

High-resolution three-dimensional graphics workstations can provide real-time representations of vehicle dynamics, control system behavior, mission execution, sonar processing and object classification. Use of well-defined, user-readable mission log files as the data transfer mechanism allows consistent and repeatable simulation of all AUV operations. Examples of integrated simulation are provided using the Naval Postgraduate School (NPS) AUV, an eight foot, 387-pound untethered robot submarine designed for research in adaptive control, mission planning, mission execution, and post-mission data analysis.

The flexibility, connectivity and versatility provided by this approach enables sophisticated visualization and analysis of all aspects of AUV development. Integrated simulator networking is recommended as a fundamental requirement for comprehensive and rapid AUV research and development.

1 Introduction

1.1 Problem Statement

Designing, building and testing an Autonomous Underwater Vehicle (AUV) is difficult. Unlike most other mobile robots, AUVs must operate unattended and uncontrolled in a remote and unforgiving environment. Inaccessibility greatly complicates evaluation, diagnosis and correction of AUV system faults. In order to ensure complete reliability, AUV software and hardware need to be fully tested in the laboratory before operational deployment. Such important testing requirements cannot be met using only a standalone AUV.

1.2 Motivation

The principal motivation driving the development of an AUV integrated simulator is to meet the research needs of the large academic group working on the Naval Postgraduate School (NPS) AUV. Students and professors have diverse research goals which are often forced to compete for access to vehicle system software and limited pool test time. The need to use operational software running on actual NPS AUV hardware is a particularly important requirement. Lack of accessibility to the NPS AUV in a distributed laboratory environment has occasionally prevented porting new software applications into the vehicle. Pre-mission validation of vehicle systems response to new software has been similarly limited in scope, resulting in several operational test failures and frustrating delays in development.

The integrated simulation approach has great value and general applicability. The NPS AUV Integrated Simulator has been designed to support complete scientific visualization of actual NPS AUV vehicle performance. The lessons learned while building this integrated simulator have proven that distributed research can be effectively accomplished when proper network connections and data-passing mechanisms are provided.

1.3 Definition and Objectives of Integrated Simulation

Integrated simulation is defined as the effective networking of a three-dimensional graphical simulation workstation with an AUV microprocessor, appropriate support equipment and all software development workstations. Integrated simulation allows coordinated utilization of computer resources for maximum realism and effectiveness. The purpose of this paper is to demonstrate the use of integrated simulation as an essential approach for rapidly designing, developing and evaluating AUVs.

An AUV integrated simulator remotely links vehicle components and support equipment with graphics simulation workstations. Networking allows complete pre-mission, pseudo-mission and post-mission visualization and analysis in a real-time lab environment. Complete integrated simulator testing of software and hardware supports prompt diagnosis and robust correction of system faults. Figure 1 shows a sample AUV integrated simulator network and primary components.

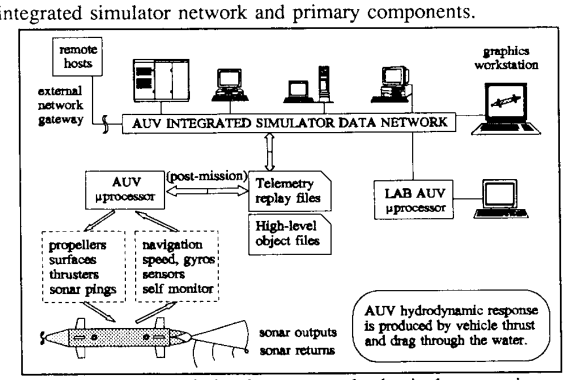

Figure 1 Integrated simulator network physical connections

Pre-mission simulator testing of AUV software permits experimentation and preliminary evaluation of developmental software. Pseudo-mission testing using an identical laboratory microprocessor or remote communication with an actual AUV permits end-to-end testing of all AUV software and hardware. Post-mission simulator playback of recorded telemetry, sonar sensor data and system state transitions supports in-depth reenactment, playback and analysis of actual operational results.

0-7803-0704-6/92$03.00 ©IEEE

1.4 Previous Work

Several graphics simulators have been previously developed at NPS to support AUV research. These simulators all operate on Silicon Graphics Inc. Iris graphics workstations.

Seow Meng Ong developed a simulator that remotely networks an Iris workstation with a Symbolics Inc. Lisp machine for real-time communication by mission planning and path planning software [20] [23]. CDR Thomas A. Jurewicz USN developed a real-time NPS AUV simulator that featured a complete hydrodynamics model and bathymetric survey terrain data of Monterey Bay [17] [24]. CDR Charles A. Floyd USN extended the Jurewicz simulator to demonstrate sonar detection and collision avoidance software [12] [13]. MAJ Ronald B. Byrnes USA and LCDR David L. MacPherson USN utilized the network capabilities of the Ong simulator to visually compare hierarchical and subsumption software architectures for AUV control [8].

Other less complex simulation methods have also been used for NPS AUV development. Most NPS AUV control system theses have analyzed vehicle performance parameters individually using mathematics support packages such as MATLAB [19], forcing researchers to visually correlate numerous two-dimensional plots of telemetry data in order to interpret test results.

Other underwater vehicle projects have also used offline graphics simulation as a design tool. As an example, C.S. Draper Laboratories has a large and sophisticated simulator which supports the development of the Defense Advanced Research Projects Agency (DARPA) Unmanned Underwater Vehicle (UUV). This simulator employs sophisticated computer models of hydrodynamic characteristics and individual physical component responses. Mission software is loaded on a separate mainframe to emulate vehicle multiprocessor response. The simulator does not incorporate actual DARPA UUV multiprocessor hardware or allow direct playback of UUV system and sensor data collected in the water. However the many capabilities of this powerful support simulator have significantly contributed to the reliability of the DARPA UUV, allowing successful and rapid progress along an ambitious development schedule [21] [15].

All of these simulation approaches successfully demonstrate the concepts they are intended to evaluate. However, none of these simulators were designed to use actual vehicle hardware or to provide general extendability to support every aspect of AUV research.

2 AUV Design and Development Considerations

AUVs are complex systems. A number of design and employment criteria unique to AUVs must be considered when determining integrated simulator specifications.

2.1 AUV Inaccessibility During Operation

The development and testing of AUV hardware and software is greatly complicated by vehicle inaccessibility during operation. AUVs are designed to operate with complete independence in an environment that makes communication and monitoring difficult. Vehicle independence design constraints leave operators unable to monitor performance, diagnose problems or override failures. This inaccessibility is perhaps the biggest liability inherent in AUV testing since it can easily lead to catastrophic failure and vehicle loss. Even when supervisory control is possible through use of a tether or underwater communications, underwater vehicle systems must be robust enough to recover and return in the event of system failures combined with communication loss. Integrated simulation can fully test fault tolerance and emergency recovery procedures of an AUV prior to risking loss of communications during independent operation.

2.2 Reliability is Paramount

Loss of an AUV due to internal failure or inability to cope with an unpredictable environment is unacceptable due to the current high cost of AUV construction and support. Furthermore if an AUV is employed in military missions such as submarine support or minefield search, human lives and operational success may depend on complete vehicle reliability. Thus the principal requirement for any AUV is that the vehicle operates dependably in all possible scenarios and under all possible failure conditions. Pre-mission verification of proper AUV performance using an integrated simulator is the only way to ensure complete vehicle integrity and verify strict reliability requirements for all software and hardware components.

2.3 Wide Variety of Software Process Types

The highly complex behaviors expected of AUVs are only possible when numerous software modules are written to handle functions such as path planning, sonar interpretation, mission control etc. Such software programs can be considered artificial intelligence (AI) applications in that human intelligence would otherwise be required to perform these challenging tasks. It is important that these high-level software modules are able to fully interact with each other for proper execution and evaluation. However such interaction is difficult when the researchers developing software are distributed over a network. Integrated simulation provides full connectivity between research software modules and the AUV microprocessor. Integrated simulation also provides data-passing mechanisms which permit interprocess communication regardless of the various host operating systems or programming languages used.

3 Integrated Simulator Software Architecture

3.1 Software Engineering Considerations

Proper design of an AUV integrated simulator addresses many requirements including repeatability, flexibility, cost-effectiveness, portability, maintainability, future growth potential and ability to upgrade. These goals can be met by following fundamental software engineering principles such as clearly defining software module specifications and functional descriptions. Formally defined data dictionary entries, data structures and spatial coordinate systems are also important. Specifications must be flexible enough to support future improvements and comprehensible enough to be rigorously followed. Frequent and thorough communication and cooperation among project members is important in order to establish formal project standards and ensure long-term success.

3.2 Integrated Simulator Software Architecture Requirements

Integrated simulator software must perform a large number of tasks. The AUV must be modeled using some simulated components (e.g. control surfaces, propellers, gyrocompass) together with actual running AUV mission software. Vehicle physical motion and behavior can be provided by the state equations of a dynamic response model. The world model should include stationary obstacles, mobile objects and the sensor interactions expected to occur as the vehicle probes the external environment model. Developmental AUV processes which have not yet been ported into the vehicle mission software need to have some way of interfacing with both the AUV microprocessor and the simulation. Operating system and programming language incompatibilities should not be an impediment to AUV software developers. Finally and perhaps most importantly, a powerful graphics workstation should render an external view of the simulated world in three dimensions with full functionality and real-time response.

3.3 Simulated and Actual Components

Maximum simulation realism is provided when actual AUV components are tested end-to-end in the laboratory. For example, an AUV might be fixed in place on blocks in a test tank while a test mission was conducted. Proper activation of sonar, rudders, diving planes and propellers would provide positive indications of correct performance. It is interesting to note that networking a test-tank AUV to a graphical simulator can give evaluators real-time insight into what the vehicle "thinks" it is doing. However, if a laboratory AUV microprocessor is used instead of the actual vehicle, the missing vehicle physical components must be separately simulated. Such simulation is accomplished by modular substitution of mathematical models for the missing physical components. A particular benefit of this approach is that AUV software testing is freed from direct interaction with the actual AUV, since the vehicle might be operating, undergoing repairs or otherwise inaccessible. The logical relationships between AUV, simulator, laboratory development network and real-world environment are shown in Figures 2 and 3.

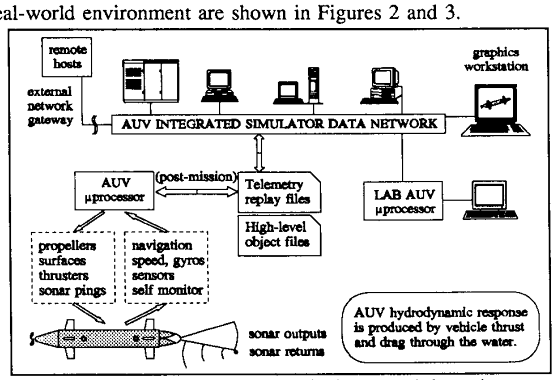

Figure 2 Integrated simulator logical connectivity using actual AUV

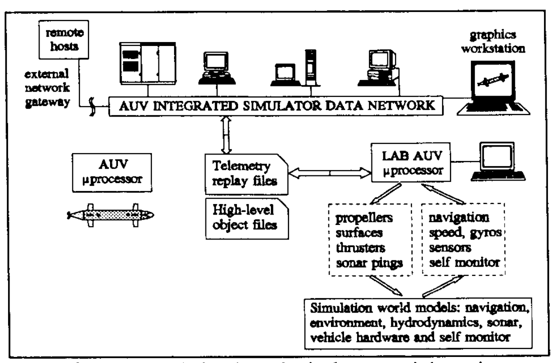

Figure 3 Integrated simulator logical connectivity using laboratory AUV

3.4 Data Transfer Mechanisms

Data transfer mechanisms are a critical component of interprocess communication. Two file types and two data transfer mechanisms are considered: telemetry replay files, high-level object files, remote file transfer and stream sockets.

In order to portray and replay AUV behavior, telemetry recorded by the vehicle must be readable by the integrated simulator. Typically such data includes vehicle position, vehicle orientation, linear and rotational velocities or accelerations, sensor data and vehicle state information, all repeated at a high data rate. Telemetry replay files can be saved by an AUV for post-mission upload or transmitted during operation. These files can also be read (with effort) by human operators, or ported as input to mathematics support packages for selective analysis of system parameters. However, Figure 4 illustrates the difficulty in portraying three-dimensional AUV track data using two-dimensional time-versus-z and x-versus-y plots [10].

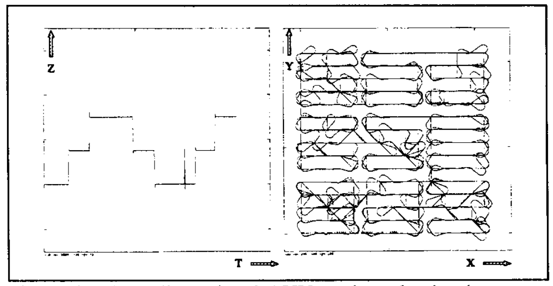

Figure 4 Three-dimensional AUV track evaluation is difficult when using multiple two-dimensional plots

Use of well-defined and consistent telemetry replay files allows repeatable simulation of all AUV missions. Telemetry replay files are also a convenient method for new mission software to record primary aspects of AUV behavior during standalone testing for later visualization on the integrated simulator. Figure 5 shows an example telemetry replay file format.

<time>	; telemetry data point time
<x> <y> <z>	; vehicle estimate of position
<φ> <θ> <ψ>	; measured 3D orientation
<p> <q> <r>	; navigation system velocities
<Δ dive> <Δ rudder>	; plane surface positions
<rpm> <log speed>	; ordered and measured speed
<sensor data fields>	; all possible sensor returns
<wildcard>	; extra slot for
	; mission-dependent use

Figure 5 Example telemetry replay file format

High-level object files allow communication of symbolic data such as position of objects, object classification, operator instructions and interprocess commands. Keeping such data in plain text makes them readable by human operators, individual AUV software processes and the integrated simulator. Optional time parameters on each command line allow high-level object files to supplement telemetry replay files for synchronized real-time playback. This combination of telemetry replay files and high-level object files allows simple and effective communication of all possible types of AUV information. An example high-level object file format is shown in Figure 6.

File transfer is the fastest and easiest way to record and communicate large amounts of data over a distributed research network. An integrated simulator network must be able to transfer telemetry replay and high-level object files between all network nodes.

Once a file transfer capability has been established, stream sockets can be implemented if transfer of individual data packets is desired [2]. Stream sockets can connect all processors on an integrated simulator network, allowing direct interprocess communication, near real-time data transfer and better evaluation

```
Environment  "worldfilename"          ; change default world
AUV          <x>   <y>   <z>          ; AUV initial position
Point        <x>   <y>   <z>          ; Position coordinates

Segment      <x1>  <y1>  <z1>  <x2>  <y2><z2>
                                      ; endpoint coordinates

Wall         <x1>  <y1>  <z1>  <x2>  <y2><z2>
                                      ; opposite corners

Cylinder     <x>   <y>   <z>   <r>   <h>
Mine         <x>   <y>   <z>   <scale>[time <t>]
                                      ; time tags optional

Ship         <x>   <y>   <z>   <scale>[time <t>]
Object       "filename"  <x>   <y>   <z> [time <t>]
Message      [time <t>]  ... free format text here ...
             ; messages can be mission log outputs or
             ; interprocess communication
```

Figure 6 Example high-level object file format

of multiple process interaction.

3.5 Distributed Artificial Intelligence Considerations

A large number of interrelated AI software processes are required for an AUV to competently perform the many behaviors required of an independent submersible. In order to keep up with demanding mission requirements, these processes must be capable of performing in real time and in parallel. Similar real-time and parallel processing support will be necessary for a graphics workstation to provide correspondingly realistic playback and interaction.

Interprocess communication and real-time process interaction are usually difficult to implement, especially if multiple user, multiple programming language or multiple operating system bottlenecks exist. The data transfer mechanisms described previously permit complete interaction among dissimilar distributed AI applications, regardless of whether these applications are internal or external to the AUV. This straightforward approach allows complete user and simulator accessibility to intermediate process outputs.

4 Three-Dimensional Graphics Simulation

High-resolution three-dimensional graphics workstations provide realistic representations of vehicle dynamics, control system behavior, mission execution, sonar processing and object classification.

4.1 Realistic Object Rendering and Real-Time Motion

The primary graphics requirement for an integrated simulator is realistic rendering and movement of virtual objects in real time. This capability is essential for visualizing an AUV's interaction with an underwater world in order to fully evaluate the proper operation of complex AUV software and hardware. Numerous graphics techniques can be used to provide a believable graphics display, ranging from drawing simple polygons to overlaying complex textures. Realistic portrayal of all objects in an underwater world allows intuitive and thorough analysis of large amounts of AUV data.

Maintaining a real-time playback capability is important for realistically rendering AUV interaction with physical objects. The graphics simulator program must be able to quickly refresh complex screens in order to visually present large amounts of data. Local empirical studies show that a 6 Hz screen update rate and input device response loop are the minimum requirements for simulator screen motion to appear smooth and realistic during operator interaction. Frame rates of 20-30 Hz may be needed for realistic illusion of rapid motion [3]. Speed can be increased and graphics pipeline loading reduced through simplified object geometry, simplification of lighting models, simulator source code optimization and graphics performance tuning techniques.

4.2 Physical Modeling

All AUV-related physical processes can be mathematically modeled with a high degree of accuracy. Vehicle physical response can be predicted using state equations, positional constraints, inverse kinematics and dynamics [1] [17] [22]. Sonar acoustic behavior can be modeled with increasingly complex levels of detail in order to meet both realism and system playback requirements [11]. Individual AUV hardware components can be simulated using control system models of transient and steady-state response. Object motion is adequately modeled using simple kinematics. Object positions can be easily updated whenever more recent correlated sonar data becomes available. In general, physical modeling is less processor-intensive than graphics rendering and adds no apparent overhead to graphics workstation response when properly parallelized [17] [24].

4.3 Sonar and Sensor Visualization

Sonar data is often difficult to visualize since acoustic beam and ray path behavior is very different from our vision-based perceptual expectations. Sonar remains the primary sensor used for intermediate and long range underwater detection. Sonar can also be quite effective when used for short range detection, object feature extraction or measurement of object characteristics such as doppler or frequency response. Color graphics visualization can portray the real-time behavior of sonar beams in three dimensions, allowing AUV designers to troubleshoot complex problems, optimize vehicle sensor performance and better understand how an AUV is interacting with the environment [4] [6] [9]. Other types of sensors such as laser rangefinders can also be displayed. Sensor visualization capabilities are valuable features for an integrated simulator.

5 Integrated Simulator Hardware Architecture

5.1 Workstation Compatibility

There are surprisingly few hardware constraints on the individual workstations making up the distributed network portion of an integrated simulator. A variety of normally incompatible operating systems and programming environments may be used as long as network connections provide a open data transfer path. Even application source code may be in a language foreign to the AUV. For example, a high-level language (e.g. Lisp, CLIPS or Prolog) may be used for rapid prototyping and initial development. Testing is then accomplished using interprocess communication and real-time data transfer of high-level object file information with the AUV. After initial process testing is complete, working high-level language code can be translated and ported into the native language of the AUV (e.g. ANSI C). This open architecture approach allows great flexibility and maximum use of available resources.

5.2 External Network Connectivity

Fully networked connections between all major support components of the AUV is essential to provide a responsive research environment. Additional external network connections will further extend AUV integrated simulator capabilities. For example, laboratory data transfers over a wide-area network or Internet allow joint AUV research over long distances. For another example, actual telemetry replay files can be transferred from a moored AUV via modem or radio link for immediate remote replay, analysis and verification. Such capabilities are particularly important when an AUV is deployed at great distances from support laboratories and immediate analysis of collected information is necessary.

6 Implementation, Evaluation and Experimental Results

An integrated simulator has been implemented for the NPS AUV [6]. This section describes the primary components and key features of the NPS AUV Integrated Simulator.

6.1 NPS AUV Vehicle Description and Sonar Characteristics

Naval officers and civilian scientists at NPS are conducting active research using an AUV designed and constructed at the school. The NPS AUV is used for basic research and thesis work in control systems technology, artificial intelligence, scientific visualization and systems integration. Specific NPS AUV project objectives include the study of mission planning, navigation, collision avoidance, real-time mission control, replanning, object recognition, vehicle dynamic motion control, and post-mission data analysis [16] [4].

The NPS AUV is eight feet long and neutrally buoyant, displacing 387 pounds with overall size and shape comparable to a small dolphin. Current vehicle endurance is two to three hours. Maximum speed of the NPS AUV is about two knots. The NPS AUV turning diameter is under three body lengths, designed to be ideal for maneuvering in the large NPS swimming pool. The NPS pool allows precise testing in a quiet, controlled environment. Open-ocean testing is feasible but is being reserved for a more robust follow-on vehicle. Video clips showing normal NPS AUV operation are available in [5] [6].

The primary components of the NPS AUV are an aluminum hull, fiberglass sonar dome, four high-frequency directional sonar transducers, twin counter-rotating four-inch propellers, lead-acid batteries, eight plane surfaces, and a Gespac computer running a Motorola 68030 processor with a 2 MB RAM card under the OS-9 operating sytem. Figure 7 shows a general schematic of the NPS AUV.

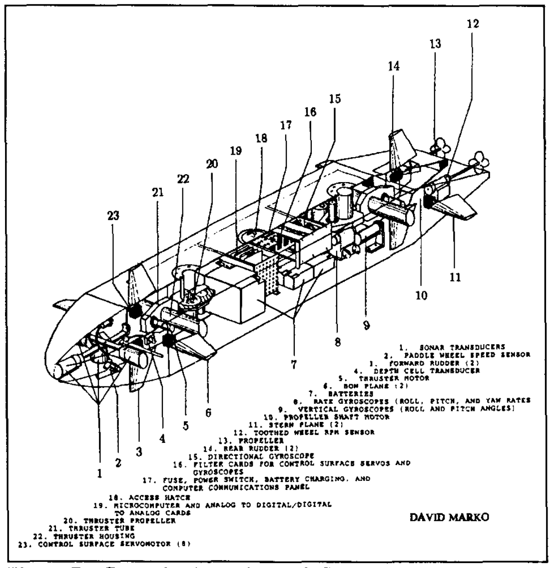

Figure 7. General schematic of NPS AUV

Four PSA-900 Programmable Sonar Altimeters made by Datasonics Inc. are orthogonally fixed in the nose of the NPS AUV pointing directly ahead, downward and to port and starboard. These transducers are fixed frequency and ultrasonic, each at approximately 200 Khz. Sonar range gate is selectable at 30 m or 300 m, and pulse length is 350 μs. Normal pulse repetition rate is 10 Hz. Sonar beamwidth is seven degrees and range resolution is 1 cm at 30 m.

6.2 NPS AUV Integrated Simulator

The NPS AUV Integrated Simulator has been developed to support NPS AUV research and demonstrate each of the concepts described in this paper [6].

High-level NPS AUV software processes are initially developed and tested on the Unix-based computer science department network. These processes can now be ported, compiled, linked and loaded on a Gespac VME-bus 68020 or 68030 microprocessor running under the OS-9 operating system. Gespac microprocessors are used both on the NPS AUV and on a separate networked laboratory AUV. The laboratory AUV includes 68020 microprocessor and I/O cards, a monitor terminal, Ethernet network connections and a networked IBM-compatible support PC which includes an OS-9 "C" language cross-compiler. The laboratory AUV also has additional hardware card slots in order to test new hardware components and new vehicle software.

Graphics simulation using the NPS AUV Integrated Simulator is just beginning to be used for laboratory evaluation of software that will run in the AUV proper. New hardware and software can be rapidly tested prior to installation and operation in the NPS AUV, minimizing vehicle risk while saving time and money. Replays of actual data recorded by the NPS AUV can be used for visualization of remote environments and detailed post-mission data analysis. Connection of software development workstations with the NPS AUV Integrated Simulator accelerates the operational deployment of high-level mission software.

The NPS AUV Integrated Simulator control panel has been written using NPS Panel Designer software, making it quickly modifiable and extendable [18]. The graphic simulator user interface permits precise control of viewpoint and reference point, lighting and rendering functions, object positions, real-time mode, high-level object file recall, individual object control and playback of telemetry replay files. The NPS AUV Integrated Simulator control panel is shown in Figure 8.

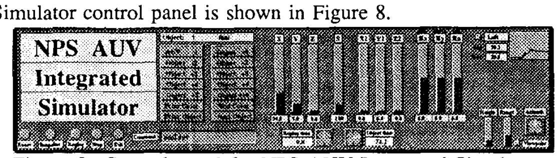

Figure 8 Control panel for NPS AUV Integrated Simulator

Because the AUV hull shape is similar to the original Swimmer Delivery Vehicle used by U.S. Navy SEAL teams, a sophisticated mathematical model is already available for simulator use to accurately recreate vehicle dynamic motion and response characteristics [17]. The hydrodynamics model state equations contain approximately 120 coefficients which continue to be improved and verified by pool testing and ongoing thesis work.

In order to display a variety of sonar data, multiple objects can be displayed on the graphics workstation. Implemented object primitives include AUV, point, line, wall, mine and cylinder. These objects can be graphically displayed simultaneously with original telemetry replay data in order to analytically visualize the validity and usefulness of various sonar classification techniques. Objects can be independently manipulated and positioned. An optional time slot for each object allows them to appear only when appropriate during synchronized playback of telemetry files. Additional advanced graphics techniques can quickly be added to the baseline graphics simulation program.

6.3 Silicon Graphics IRIS Workstation Capabilities

The NPS AUV Integrated Simulator uses a Silicon Graphics Inc. Iris 4D/240VGX. This graphics workstation has 48 bit color, 24 bit Z-buffering and four parallel 25 Mhz 20 MIPS processors which together can process 1 M vectors, 1.1 M triangles or 180 K polygons per second [14]. Other slower IRIS workstations are also available for use, including a remote workstation in the Mechanical Engineering Department adjacent to the NPS AUV support laboratory. All graphics workstations are connected by local or wide area networks. Real-time playback of telemetry data is automatically adjusted to take maximum advantage of the current graphics workstation processing power, producing realistic screen displays regardless of which model graphics workstation is used.

6.4 Laboratory AUV Simulation

A primary objective of integrated simulation is to run operational software on a laboratory version of the AUV microprocessor. The NPS AUV Integrated Simulator includes an identical Gespac computer running a Motorola 68030 under the OS-9 operating sytem. Added to this computer are interface cards for a VT220 monitor and keyboard for external control, serial connection to a PC, and Ethernet connection to the NPS computer science department network. Full connectivity is thus provided to all developmental workstations of interest as well as the Campus Wide Network and Internet. Since OS-9 is a multiprocess real-time system, multiple users can access the Gespac AUV microprocessor simultaneously.

Unmodified operational NPS AUV software is able to run successfully on the laboratory AUV microprocessor, and telemetry data files are properly saved during each run. Telemetry data files have been successfully transferred over the network and played back on the Iris graphics workstation. Although missing NPS AUV hardware such as sonar and plane surface response has not yet been simulated, successful visualization of the laboratory test runs has proven the feasibility of the integrated simulation approach. Functional AUV software and hardware is now directly available to all NPS AUV researchers for experimentation and evaluation prior to in-water testing.

7 Additional Applications

Several applications were implemented concurrently with the NPS AUV Integrated Simulator that successfully demonstrate the usefulness of integrated simulation in support of high-level AUV-related AI research.

7.1 Sonar Classification Application

The NPS AUV Sonar Classification System uses outputs from simple active sonars to classify detected underwater objects [6] [7]. Figure 9 shows sample sonar classifications in the NPS pool displayed using the NPS AUV Integrated Simulator. Scientific visualization techniques permitted rapid and precise development of geometric analysis techniques and classification heuristics, resulting in successful completion of the NPS AUV Sonar Classification System.

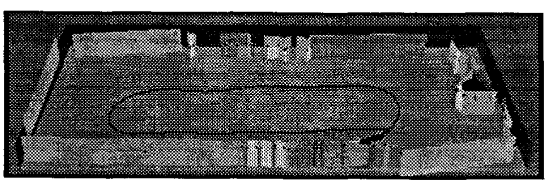

Figure 9. Integrated simulator screen display of the NPS pool, AUV track and active sonar classifications

7.2 Circle World Path Planning Application

Optimal path planning is an important area of AUV research. Displaying and viewing paths and obstacles without restrictions allows the algorithm designer to evaluate his results in the most comprehensive and challenging manner possible. Additionally, subtle difficulties which might be obscured by two-dimensional projections are clearer and easier to evaluate when shown in three-dimensions. A circle world path planner has been developed that finds shortest paths around circular or cylindrical obstacles [6]. Figure 10 shows how shortest path planning results can be portrayed in three dimensions using the NPS AUV Integrated Simulator.

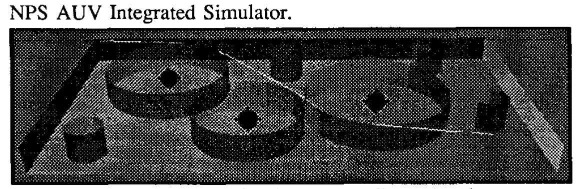

Figure 10 Integrated simulator three-dimensional representation of circle world obstacles and shortest path in the NPS pool

7.3 Minefield Search Application

Another application benefiting from integrated simulation is an AUV minefield search planner [10]. A three-dimensional open-ocean minefield model is optimally searched and mapped using a dynamic search strategy. AUV search track and vehicle posture are recorded in a simplified telemetry replay file, while waypoint objectives and detected mines are recorded in a separate high-level object file. Synchronized playback of these files allows complete visualization of the complex path taken by the AUV as well as the numerous objects detected, shown in Figure 11. Note that vehicle track is much easier to visualize than in Figure 4, particularly since the simulator user's viewpoint can be panned over and around the track data.

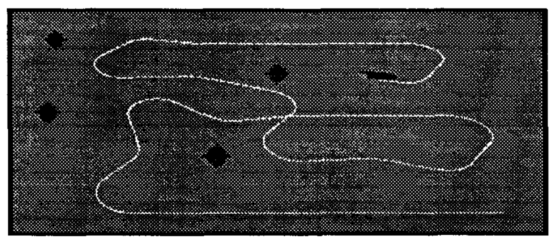

Figure 11 Integrated simulation display of AUV minefield search

8 Additional Applicability, Limitations and Future Work

8.1 Comparison of Theoretical and Empirical Data

Three-dimensional visualization techniques are well suited for making meaningful comparisons between large abstract data sets. Such comparisons can significantly aid the operator in evaluating small errors in mathematical models or system control software. For example, predicted vehicle track for a given control systems algorithm could be spatially superimposed over actual test track data. Coefficients in the prediction model can then be incrementally adjusted until theoretical behavior matches actual performance. A similar visualization approach was used with great success while determining precise heuristics for object classification using recorded active sonar data [6] [7]. Direct comparison of theoretical and empirical data is a powerful diagnostic tool that can be used to improve thoeretical formulations as well as vehicle implementations.

8.2 Limitations to Integrated Simulation

The primary limitation on integrated simulation realism is graphics workstation speed and capability. Many graphics workstations can generate photorealistic images but are unable to rapidly reproduce a series of images in real time. The competing requirements between rendering accuracy and adequate frame rate will always require design tradeoffs by the graphics programmer. Silicon Graphics Inc. workstations use the GL Graphics Library, which is a good graphics programming choice due to the numerous graphics techniques provided, code optimization, portability to other platforms and open licensing availability. Graphics workstation capabilities are probably the most critical consideration in integrated simulator design.

The local area network (LAN) used to connect integrated simulator nodes should be reliable, have adequate throughput and allow addition or removal of nodes with little difficulty. Ethernet-based LANs are adequate for NPS AUV Integrated Simulator requirements and also provide gateway connectivity to Internet. It should be noted that under most network protocols socket stream packet delivery order is not guaranteed and timing of packet delivery is somewhat unpredictable. Processes that use socket stream data should be flexible and not tied to hard real-time requirements.

Computer security is a consideration if sensor data or mission software is proprietary or classified. The use of plain text for telemetry replay files and high-level object files permits the use of encryption protocols during transfer. Encrypting files is a simple technique that imposes minimal processing overhead. Individual nodes on the integrated simulator network will require standard security precautions against unauthorized remote access.

8.3 Future Use of Integrated Simulation

Integrated simulation provides development benefits to all types of remote vehicles, regardless of whether a communications tether is present or remote control by human operators is required. Integrated simulation not only solves a number of the problems that degrade robot implementation, but also provides tools to work on practical system engineering and integration problems which previously were too difficult to address. The authors hope that widespread incorporation of integrated simulation techniques will improve the accessibility, intelligibility and progress rate of mobile robot research.

9 Conclusions

Integrated simulation allows all AUV systems to be tested in a timely and complete manner. The flexibility and connectivity provided by this approach enables sophisticated visualization and complete analysis of all aspects of AUV development. Integrated simulator networking is recommended as a fundamental requirement for comprehensive and rapid AUV research and development.

Acknowledgements

The authors would like to thank CAPT Alan R. Beam USN of DARPA and Mr. Patrick Hale of Charles S. Draper Laboratories for access to the DARPA UUV and the UUV support simulator. The authors also wish to thank Dr. Anthony M. Healey and Dr. Robert B. McGhee for their continuing guidance and inspiration.

References

[1] Badler, Norman I., Barsky, Brian A. and Zeltzner, David, ed., *Making Them Move: Mechanics, Control and Animation of Articulated Figures*, Morgan Kaufmann Publishers Inc., San Mateo, California, 1991.

[2] Barrow, T.H., Yurchak, J.M. and Zyda, M.J., *Distributed Computer Communications In Support Of Real-Time Visual Simulations*, M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1988.

[3] Brooks, Frederick P. Jr., "Grasping Reality through Illusion: Interactive Graphics Serving Science," included in "Implementing and Interacting with Real-time Microworlds," course 29, *ACM SIGGRAPH Conference*, Boston, Massachusetts, 31 July-4 August 1989, pp. 3-1 through 3-11.

[4] Brutzman, Donald P. and Compton, Mark A., "AUV Research at the Naval Postgraduate School," *Sea Technology*, vol. 32 no. 12, December 1991, pp. 35-40.

[5] Brutzman, Donald P., Floyd, Charles A. and Whalen, Russell, "Naval Postgraduate School Autonomous Underwater Vehicle," *Video Proceedings of the IEEE International Conference on Robotics and Automation 92*, Nice, France, May 1992.

[6] Brutzman, LCDR Donald P., *NPS AUV Integrated Simulator*, Master's Thesis, Naval Postgraduate School, Monterey, California, March 1992.

[7] Brutzman, LCDR Donald P., Compton, LCDR Mark A. and Kanayama, Yutaka, "Autonomous Sonar Classification using Expert Systems," draft article, *OCEANS 92* conference, Oceanic Engineering Society of the IEEE, Newport, Rhode Island, October 26-29, 1992.

[8] Byrnes, R.B., MacPherson, D.L., Kwak, S.H., Nelson, M.L. and McGhee, R.B., "An Experimental Comparison of Hierarchical and Subsumption Software Archtectures for Control of an Autonomous Underwater Vehicle," presented at IEEE Oceanic Engineering Society Symposium on Autonomous Underwater Vehicles, Washington D.C., June 2-3, 1992.

[9] Compton, LCDR Mark A., *Modeling the Sonar Environment*, unpublished paper, Naval Postgraduate School, Monterey, California, September 1991.

[10] Compton, LCDR Mark A., *Minefield Search and Object Recognition for Autonomous Underwater Vehicles*, Master's Thesis, Naval Postgraduate School, Monterey, California, March 1992.

[11] Etter, Paul C., *Underwater Acoustic Modeling: Principles, Techniques and Applications*, Elsevier Applied Science, London, England, 1991.

[12] Floyd, Charles A., Kanayama, Yutaka, and Magrino, Christopher, "Underwater Obstacle Recognition using a Low-Resolution Sonar," *Proceedings of the Seventh International Symposium on Unmanned Untethered Submersible Technology*, University of New Hampshire, Durham, New Hampshire, September 1991, pp. 309-327.

[13] Floyd, Charles A., *Design and Implementation of a Collision Avoidance System for the NPS Autonomous Underwater Vehicle (AUV II) Utilizing Ultrasonic Sensors*, Master's Thesis, Naval Postgraduate School, Monterey, California, September 1991.

[14] Gorey, Kevin, "Periodic Table of the Irises," Silicon Graphics Inc., Mountain View, California, February 1991.

[15] Interview with Patrick Hale, DARPA UUV Project Manager, C.S. Draper Laboratories, Cambridge, Massachusetts by the author, December 11, 1991.

[16] Healey, A.J., McGhee, R.B., Cristi, R., Papoulias, F.A., Kwak, S.H., Kanayama, Y., and Lee, Y., "Mission Planning, Execution, and Data Analysis for the NPS AUV II Autonomous Underwater Vehicle", *Proceedings of the First IARP Workshop on Mobile Robots for Subsea Environments*, Monterey, California, October 23-26, 1990, pp. 177-186.

[17] Jurewicz, CDR Thomas A., *A Real Time Autonomous Underwater Vehicle Dynamic Simulator*, Master's Thesis, Naval Postgraduate School, Monterey, California, December 1990.

[18] King, LT David Maurice and Prevatt, LCDR Richard Montgomery III, "Rapid Production of Graphical User Interfaces," Master's Thesis, Naval Postgraduate School, Monterey, California, December 1990.

[19] Mathworks, Inc., *PC-MATLAB for MS-DOS Personal Computers*, South Natick, Massachusetts, 1989.

[20] Ong, Seow Meng, *A Mission Planning Expert System with Three-Dimensional Path Optimization for the NPS Model 2 Autonomous Underwater Vehicle*, Master's Thesis, Naval Postgraduate School, Monterey, California, June 1990.

[21] Pappas, George, Shotts, William, O'Brien, Mack and Wyman, William, "The DARPA/Navy Unmanned Undersea Vehicle Program," *Unmanned Systems*, vol. 9 no. 2, Spring 1991, pp. 24-30.

[22] Thalmann, Nadia Magnenat, and Thalmann, Daniel, *Computer Animation: Theory and Practice*, second edition, Springer-Verlag, Tokyo, Japan, 1990.

[23] Zyda, M.J., McGhee, R.B., Kwak, S.H., Nordman, D.B., Rogers, R.C. and Marco, D., "Three-Dimensional Visualization of Mission Planning and Control for the NPS Autonomous Underwater Vehicle," *IEEE Journal of Oceanic Engineering*, vol. 15 no. 3, July 1990, pp. 217-221.

[24] Zyda, Michael J., Jurewicz, Thomas A., Floyd, Charles A. and McGhee, Robert B., "Physically Based Modeling of Rigid Body Motion in a Real-Time Graphical Simulator," unpublished paper, Naval Postgraduate School, Monterey, California, September 1991.